



TITLE:

DKA法による固有値計算の試み(並列数値計算アルゴリズムとその周辺)

AUTHOR(S):

戸川, 隼人

CITATION:

戸川, 隼人. DKA法による固有値計算の試み(並列数値計算アルゴリズムとその周辺). 数理解析研究所講究録 1986, 585: 273-274

ISSUE DATE:

1986-02

URL:

<http://hdl.handle.net/2433/99362>

RIGHT:

DKA法による固有値計算の試み

日大・理工 戸川隼人 (Hayato Togawa)

要 約

高次代数方程式の解法として知られるDKA法を固有値の計算に応用することを考え、実験してみたところ、速度、精度の両面で成績がよく、実用解法として使えることがわかった。さらに、これを Lanczosの双直交化アルゴリズムと組合せると、非対称行列の固有値問題の解法として、きわめて有効であることがわかった。この方法はスパースな問題に適している。

1. DKA法

Durand と Kerner は、代数方程式

$$f(z) = z^n + a(1)z^{(n-1)} + a(2)z^{(n-2)} + \cdots + a(n-1)z + a(n) = 0$$

の n 個の根を同時に求めるために

$$\text{新 } z(i) = z(i) - \frac{f(z(i))}{\prod_{i \neq j} (z(i) - z(j))} \quad (i=1, 2, \dots, n)$$

という反復式を提案した。この方法は大域的収束性を有し、また、全部が単根ならば最終局面では2次収束となるので、近年広く使われている。

2. 固有値計算への応用

行列の固有値問題

$$A x = \lambda x$$

を解くには、

$$\det(\lambda I - A) = 0$$

を解けばよいから、反復式は

$$\text{新 } \lambda(i) = \lambda(i) - \frac{\det(\lambda(i) \cdot I - A)}{\prod_{i \neq j} (\lambda(i) - \lambda(j))} \quad (i=1, 2, \dots, n)$$

となる。ここで計算上の問題点として、行列式の値の計算におけるオーバーフローおよび精度低下が懸念されるかもしれないが、実験の結果ほとんど問題ないことがわかった。出発値としては、Aberth の考え方を踏襲して、複素平面上で全部の固有値を含む円を描き、円周上の n 等分点を出発値にするというのが1案であるが、Gerschgorin 円板を

用いる方が得策であろう。

3. 前処理

行列式の計算を能率よく行うために、あらかじめ相似変換によってAを3重対角化する。そのための算法としては、Lanczos 法が適している（3重対角化と同時に減次の効果があるため）。

[Lanczos 法による3重対角化]

2本の出発ベクトル

$$x(0), \quad y(0) \quad (\text{カッコ内の数字は反復回数})$$

を適当に選び（ほとんど任意にとれる）、下記の手続きを反復する。

$$\begin{aligned} x(k+1) &= Ax(k) - \alpha(k)x(k) - \beta(k-1)x(k-1) \\ y(k+1) &= By(k) - \alpha(k)y(k) - \beta(k-1)y(k-1) \end{aligned}$$

但し

$$\begin{aligned} B &= \text{transpose}(A) \\ \alpha(k) &= \frac{(y(k), Ax(k))}{(y(k), x(k))} \\ \beta(k) &= \frac{(x(k), y(k))}{(x(k-1), y(k-1))} \end{aligned}$$

結果は次のような3重対角行列になる。

$$T = \begin{bmatrix} \alpha(1) & \beta(1) & & & 0 \\ 1 & \alpha(2) & \beta(2) & & \\ & 1 & \alpha(3) & \beta(3) & \\ & & \ddots & \ddots & \ddots \\ 0 & & & 1 & \alpha(n) \end{bmatrix}$$

4. 行列式の計算法

行列 $(\lambda I - T)$ の左上 k 行 k 列までの小行列式を $d(k)$ とすれば、

$$d(k+1) = (\lambda - \alpha(k+1)) \cdot d(k) + \beta(k) d(k-1)$$

であるから、 $d(1) = (\lambda I - \alpha(1))$ から出発して、 $k=1, 2, 3, \dots$ の順に計算すれば、

$$\det(\lambda I - T) = d(n)$$

が求められる。